

Data Wrangling with Python Reading

Before starting any data manipulation work in a notebook environment such as Google Colaboratory or Jupyter Notebooks, we usually import the python libraries for data science and analysis. With Python, we import pandas for data manipulation and NumPy for scientific computing.

1. Reading Data

- During this step, we load our data files originally stored in many formats, ie. .csv files, SQL tables, excel files, tsv files, etc.
- We normally load our dataset using functions provided by the pandas library i.e. `read_csv()` for CSV files, `read_excel()` for excel files, etc.
- While using pandas, the stored dataset in a two-dimensional data structure is called a data frame.

2. Data Exploration

- The purpose of this step is to understand the structure of our dataset.
- We use the **head()** function provided by pandas to read the first five records.
- We use the **tail()** function to read the last records.
- We can also use the **sample()** function to get a sample of records.

3. Missing Data

- Missing data are problematic because most statistical procedures require a value for each variable. When a data set is incomplete, there is a limitation to further analysis.
- We categorize missing data in the following ways;
 - Missing completely at random (MCAR) - Missing values randomly distributed across all observations.
 - Missing at random (MAR) - Missing values not randomly distributed across observations but where missingness can be fully accounted for by variables containing complete information.
 - Missing not a random (MNAR) - Missing values on a variable are related to the values of that variable itself, even after controlling for other variables. For example, when data are missing on IQ, only people with low IQ values have

missing observations for this variable.

- There are many methods of dealing with missing data, some common of which are as follows;
 - Listwise deletion - Removing all data for an observation that has one or more missing values.
 - Dropping insignificant variables/features.
 - Mean, Median, and Mode Imputation.
 - Regression imputation.

4. Outliers

- When we work with data, we usually check our data for outliers. Outliers are data points that differ significantly from other observations.
- Outliers result from:
 - Genuine extreme high and low values in the dataset.
 - Human error.
 - Sampling error, data processing error, etc.
- After identifying the outliers, we can take several approaches to handle them:
 - We can replace the outlier with the maximum or minimum value. Since we do not know the actual value of the outlier, this approach is often not reliable.
 - A conservative approach is to replace the outlier with the mean of the process variable.
 - The third approach is to treat them as missing data.

5. Duplicate Data

- Duplicated data means repeated observations.
- Causes of duplicate data:
 - Lack of data standardization.
 - Recordkeeping issues.
- Techniques to prevent duplications of data;
 - Standardize contact data that is used in an organization by many entities.
 - Define the level of matching such that there is a tolerance level that is considered a duplicate from the system during data entry.
- We typically remove duplicate data, or if the duplicate data is a segment of existing records, then it's usually merged.

6. Filtering

- We use filtering functions to:
 - Retrieve the records relevant to our analysis.
 - Look at results for a particular period.
 - Calculate results for particular groups of interest.
 - Exclude erroneous or "bad" observations from an analysis.
 - Train and validate statistical models.
- Filtering this data includes:
 - Coming up with a rule for the observations needed.
 - Selecting the observations that fit the rule.
 - Conducting the analysis using only the information contained in those selected observations.
- Pandas will provide us with many methods for filtering; however, in our case will only cover a few.

7. Sorting

- Like filtering, we sort items in our data frame to retrieve records in a specific structure required to fulfill our analysis objectives.
- We can use the `sort_values` function of pandas with other methods to get a specific outcome, i.e., filtering and sorting our values.
- One of the significant applications of sorting is data cleaning, which is sorting data to look for abnormalities in a data pattern. Say, if we wanted to sort variable expenses over one month to look for variances.
- We can also use sorting for ranking or prioritizing records, i.e., to determine the several highest or lowest records in a variable.
- Something to watch while sorting is ensuring that our variable contains values in the respective data type, i.e., if we were to sort numerical data in a variable with the text data type, we would sort our data in alphanumerical order than numeric. That may not be our desired outcome.

8. Splitting, Merging, and Concatenation

- Whenever we would like to manipulate our data frame, especially while answering our research question, splitting, merging, and concatenation functions provided by pandas come into use.
- We can split a variable/feature when we would like to use a subset of the values contained in the variable/feature.
- We can also concatenate features with values in string format whenever we would like to use the outcome of our operation.
- In addition, the merging function provided by pandas allows us to merge two data

frames.

9. Exporting Data

- Lastly, the ability to export data allows us to store a clean dataset for further future analysis.
- We can export pandas data frames to CSV, Excel, SQL, JSON, etc. formats through the given functions.

Source: [\[Link\]](#)